REMARKS

Claims 1-4, 7-29, and 33 are currently pending in the subject application and are presently under consideration. Claims 1, 7, 15, 16, 18-22, 25, 28 and 29 have been amended as shown on pages 2-8 of Reply. Applicants' representative thanks the Examiner for the teleconference of February 24, 2009 wherein statutory subject matter issues and merits of the claims vis-à-vis Schaefer in view of Dewhurst *et al.* were discussed. The Examiner conveyed that the amended claims overcome the rejection under 35 U.S.C. §101 but necessitate further search.

Favorable reconsideration of the subject patent application is respectfully requested in view of the comments and amendments herein.

I. Rejection of Claims 1-4, 7-21, and 28-29 Under 35 U.S.C. §101

Claims 1-4, 7-21, and 28-29 stand rejected under 35 U.S.C. §101 because the claimed invention is directed to non-statutory subject matter. Withdrawal of this rejection is requested in view of the amendments to independent claims 1, 28 and 29. Such amendments are believed to place the claims within the bounds of statutory subject matter in accordance with 35 U.S.C. §101.

II. Rejection of Claims 1, 4, 7-11, 14-22, 25, 27-29, and 33 Under 35 U.S.C. §103(a)

Claims 1, 4, 7-11, 14-22, 25, 27-29, and 33 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Schaefer (US 2003/0084429) in view of Dewhurst, *et al.* (US 6,430,609). Withdrawal of this rejection is requested since Schaefer and Dewhurst, *et al.* fail to teach or suggest all aspects of subject claims.

> A factfinder should be aware, of course, of the distortion caused by hindsight bias and must be cautious of arguments reliant upon *ex post* reasoning. See *KSR v. Teleflex*, 550 U.S. ___, 127 S. Ct. 1727 (2007) citing Graham v. John Deere Co. of Kansas City, 383 U. S. 1, 36 (warning against a "temptation to read into the prior art the teachings of the invention in issue" and instructing courts to "'guard against slipping into the use of hindsight'" (*quoting Monroe Auto Equipment Co. v. Heckethorn Mfg. & Supply Co.*, 332 F. 2d 406, 412 (CA6 1964))).

Applicants' claimed subject matter relates to system and methodology to facilitate navigation for inexperienced and experienced programmers to create user interface automation and to facilitate a modular system which can be modified without recompilation of the executables. To this end, independent claim 1 recites ***a map information store comprising program flow information wherein one or more section names divide the map information store into specific page data such that each of the section name references a specific page of the automation component; a command information store disparate from the map information store that comprises the section names associated with the specific page of the automation component and one or more actions to execute for the specific page; the navigation component facilitates simulating the user interface based at least upon information stored in the command information store and the map information store, wherein the map information store, the command information store and associated executables are stored separately…***, Independent claim 22 recites:… ***storing information related to specific pages of the automation system and corresponding acts to execute for the pages in a disparate map information store and a command information store respectively; receiving mapping information from the map information store comprising one or more section names that reference specific pages of the automation system; receiving command information from the command information store comprising specific section names corresponding to information stored in the map information store and information associated with commands to be executed for respective pages of the automation system; …modifying the user interface automation utilizing existing executables upon modification of one or more of the map information store or the command information store, by storing data, commands associated with generating the user interface and the executables separately***. Similarly independent claim 25 recites: ***retrieving mapping information from a map file comprising at least a section name and a page identifier for pages generated by the automation component; retrieving commands to be executed for the pages from a command file;…. modifying the user interface automation when one or more of the command file or map file are modified_maintaining compilation of executables…*** Independent claims 28 and 29 also recite similar aspects. Schaefer and Dewhurst, *et al.*, alone or in combination fail to teach or suggest such claimed aspects.

Schaefer provides systems and methods for table driven automation testing of a software program. The system includes a GUI translator component to translate one or more GUI maps into a set of database tables, a data input component to facilitate entry and editing of test case data in the tables, and a test engine component for executing the software program based on a test case stored in the tables. Users generate GUI maps for each GUI of a software program to be tested. Each GUI map includes hierarchically organized information about a window and objects on the window, such as text fields, boxes, buttons, menus, etc. A GUI map can be generated manually by entering information about a window and the objects on the window into a text file or it can be generated utilizing a GUI map editor. The GUI maps are translated into a set of tables in a database to facilitate functional testing of a software program based on data residing in the tables (*See* Schaefer paragraph [0048]). In fact, Fig. 3 of Schaefer discloses a process of generating GUI maps for different windows. (*See* Schaefer description related to Fig. 3 at paragraph [0057]). Thus, Schaefer relates to a procedure that is converse of the process recited in subject claims. For example, independent claim 25 recites among other features: ***modifying the user interface automation when one or more of the command file or map file are modified while maintaining compilation of executables***. Schaefer discloses that any changes to the GUI of the software necessitate corresponding changes to the GUI maps. (*See* Schaefer paragraph [0093]). Nowhere does Schaefer teach or suggest that changes to the GUI maps are reflected in the GUI of the software. At page 4 of Office Action, it is erroneously asserted that Schaefer teaches *storing the map information store, the command information store and executables separately*. The Office Action submits that Schaefer shows a plurality of components individually addressable stored in memory and thus provides a means of storing components separately in memory and secondary storage. Applicants' representative disagrees and respectfully submits that Schaefer clearly shows a typical conventional system wherein the GUI map includes both program flow and command execution information (*See* Schaefer Fig. 4 and related description at paragraph [0059]). Further, on page 8 of the subject Office Action, with respect to the rejection of claim 15, it is erroneously contended that Schaefer teaches a map information store as recited in claim 1. At the cited portion, Schaefer discloses an example of a test data table mapped to a Function_Prototype table used to store a set of predefined function calls that may be used as pre-actions or post-actions during the software testing (*See* Schaefer paragraphs [0076]- [0077]). In contrast, the claimed subject matter provides for an architecture which separates data, command(s) and the executable(s), thus facilitating a modular system which can be modified without recompilation of the executable(s) (*See* applicants' specification as filed page 2 lines 1-9). The Office Action acknowledges that the primary reference, Schaefer, does not teach the claimed subject matter in its entirety with respect to *modifying the user interface automation utilizing existing executables upon modification of one or more of the map information store or the command information store, by storing data, commands and the executables separately* and provides a secondary reference, Dewhurst, *et al.* to compensate for the deficiencies of Schaefer.

Dewhurst provides for a method for configuring and executing a software application with a client user interface and fails to make up for the aforementioned deficiencies of Schaefer with respect to the claimed map and control information stores that facilitate storing program flow, data and executables separately. In accordance with Dewhurst, *et al.,* a master configuration file containing an array of configuration variables controlling the execution of software application is initially generated. A subset of configuration variables from the array of configuration variables contained in the master configuration file is pre-selected by an expert user. A client user interface (accessible to novice users) is used to modify only the subset of configuration variables by the novice users (*See* Dewhurst, *et al.* Col. 4, lines 33-65). An expert user generates a template file comprising a subset of configuration variables pre-selected from the master configuration file for modification by novice users. The novice users access the template files, change default values and transmit the new values to a computational server (*See* Dewhurst, *et al.* Col. 11, lines 5-65). Thus, Dewhurst, *et al.* discloses a template file containing all the configuration information together with the instructions for displaying user interface on the client computer. (*See* Dewhurst, *et al.* col. 11 lines 49-58). Hence, it can be concluded that Dewhurst, *et al.* fails to teach or suggest *storing data, commands and the executables separately* as recited in the subject claims. Therefore, it follows that Dewhurst, *et al.* fails to make up for the aforementioned deficiency of Schafer with respect to teaching or suggesting a map information store and a disparate command information store as recited in the subject independent claims.

A beneficial synergy is obtained by storing updateable information separate from executables that are maintained in a compiled form. For instance, the updateable information can be stored in simple text files separate from executables associated with program flow to mitigate a likelihood that that executables need to be modified and/or recompiled based on changes to the updateable information. Thus, modification of the text files produces new behavior for the claimed systems and methods; and update to the program flow only requires a modification to these text files and not the executables or engine (*See* applicants' specification as filed page 5 line 10 – page 6 line 5). In addition, the map information store includes section name(s) that divide the map information store into specific page data where a specific section name references a specific page of an associated automation component (*e.g.,* wizard). (*See* applicants' specification as filed page 7 lines 10-15). A disparate command information store stores command information associated with the automation component. It contains the information about which page to look for and what action(s) to execute for a given page. (*See* applicants' specification as filed page 8 lines 4-6). Thus, section name 210 [Welcome Page] in the map information store illustrated in applicants' Fig. 2 corresponds to command 310, [Welcome Page\30] in the command information store illustrated in Fig. 3. Similarly, "[Welcome?\60]" from a command information store can correspond to a section in the map information store (*e.g.,* mapping file) also named "Welcome". The "?" indicates that this page is optional and the automation should continue on to the next section if it does not encounter the page within 60 seconds. Section data 320 is associated with section name(s) 310. Generally, the section data is of the form "command = data". (*See* applicants' specification as filed page 8 lines 24-28). Schaefer and Dewhurst neither teach nor suggest such separation of mapping and command information.

In view of at least the foregoing, it is clear that Schaefer and Dewhurst fail to teach or suggest to one of ordinary skill in the art each and every aspect recited in independent claims 1, 22, 25, 28 and 29, and dependent claims 4-11, 14-21, and 29. Therefore, it is respectfully requested that this rejection be withdrawn.

III. Rejection of Claim 2 Under 35 U.S.C. §103(a)

Claim 2 stands rejected under 35 U.S.C. §103(a) as being unpatentable over Schaefer (US 2003/0084429) in view of Dewhurst, *et al.* (US 6,430,609) in further view of Minard (US 6,247,020). It is respectfully requested that this rejection be withdrawn for at least the following reasons. Schaefer, Dewhurst, *et al.* and Minard either alone or in combination do not teach or suggest all aspects set forth in the subject claims. Minard relates to development system with application browser user interface and does not make up for the aforementioned deficiencies of Schaefer and Dewhurst, *et al.* with respect to *the navigation component facilitates simulating the* *user interface based at least upon information stored in the command information store and the map information store, wherein the map information store, the command information store and associated executables are stored separately, the navigation component further modifies the user interface automation without recompiling executables upon modification of one or more of the map information store or the command information store,* as recited in independent claim 1 (from which claim 2 depends). Thus it is submitted, the subject matter as recited in claim 2 is not obvious over the combination of Schaefer, Dewhurst, *et al.* and Minard. Accordingly, it is respectfully submitted that this rejection should be withdrawn.

IV. Rejection of Claims 12-13, 23, and 26 Under 35 U.S.C. §103(a)

Claims 12-13, 23, and 26 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Schaefer (US 2003/0084429) in view of Dewhurst, *et al.* (US 6,430,609) in further view of Zimniewiez, *et al.* (US 6,744,450). It is respectfully requested that this rejection be withdrawn for at least the following reasons. Schaefer, Dewhurst, *et al.* and Zimniewiez, *et al.* either alone or in combination do not teach or suggest all aspects set forth in the subject claims. Zimniewiez, *et al.* relates to providing multiple installation actions and does not make up for the deficiencies of Schaefer and Dewhurst, *et al.* with respect to *the navigation component facilitates simulating the user interface based at least upon information stored in the command information store and the map information store, wherein the map information store, the command information store and associated executables are stored separately, the navigation component further modifies the user interface automation without recompiling executables upon modification of one or more of the map information store or the command information store,* as recited in independent claim 1 (from which claims 12 and 13 depend), and similarly by independent claim 22 (from which claim 23 depends) and independent claim 25 (from which claim 26 depends). Thus it is submitted, the claimed subject matter as recited in claims 12, 13, 23 and 26 is not obvious over the combination of Schaefer, Dewhurst, *et al.* and Zimniewiez, *et al.* Accordingly, it is respectfully submitted that this rejection should be withdrawn.

CONCLUSION

The present application is believed to be in condition for allowance in view of the above comments and amendments. A prompt action to such end is earnestly solicited.

In the event any fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063 [MSFTP462US].

Should the Examiner believe a telephone interview would be helpful to expedite favorable prosecution, the Examiner is invited to contact applicants' undersigned representative at the telephone number below.

Respectfully submitted,
TUROCY & WATSON, LLP

/Bhavani Rayaprolu/
Bhavani Rayaprolu
Reg. No. 56,583

TUROCY & WATSON, LLP
57TH Floor, Key Tower
127 Public Square
Cleveland, Ohio 44114
Telephone (216) 696-8730
Facsimile (216) 696-8731